AF/2183

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appellant:	**Andrea OLGIATI et al.**	Examiner:	Tonia L. MEONSKE
Serial No.:	**09/762,981**	Art Unit:	2183
Filed:	May 9, 2001	Our Ref:	B-4089PCT 618533-7 30990081-4 US
For:	"COMPUTER ARCHITECTURE CONTAINING PROCESSOR AND COPROCESSOR"	Date:	February 9, 2007
		Re:	***Appeal to the Board of Appeals***

BRIEF ON APPEAL

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This is an appeal from the Final rejection dated February 6, 2006, for the above identified patent application. This Appeal Brief is being timely filed in reply to the Notification of Non-Compliant Appeal Brief mailed on January 12, 2007, a response to which is initially due by February 12, 2007, and is intended to replace the Appeal Brief previously submitted on August 7, 2006 in support of the Notice of Appeal filed on June 7, 2006. If necessary, please deduct the amount of $500.00 for the fee set forth in 37 C.F.R. 1.17(c) for submitting this Brief from deposit account no. 08-2025.

REAL PARTY IN INTEREST

The real party in interest to the present application is Hewlett-Packard Development Company, LP, a limited partnership established under the laws of the State of Texas and having a principal place of business at 20555 S.H. 249 Houston, TX 77070, U.S.A. (hereinafter "HPDC"). HPDC is a Texas limited partnership and is a wholly-owned affiliate of Hewlett-Packard Company, a Delaware Corporation, headquartered in Palo Alto, CA. The general or managing partner of HPDC is HPQ Holdings, LLC.

RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences related to the present application.

STATUS OF CLAIMS

Claims 1-12, 14, 15, 17 and 18 are the subject of this Appeal and are reproduced in the accompanying appendix.

STATUS OF AMENDMENTS

No Amendment After Final Rejection has been entered.

SUMMARY OF CLAIMED SUBJECT MATTER

The invention claimed in claim 1 is directed to a computer system comprising a first processor 1 (p. 5 l. 32; Fig. 1); a second processor 2 for use as a coprocessor to the first processor (p. 5 l. 32 – p. 6 l. 1); a coprocessor controller 9 (p. 6 ll. 20-21; Fig. 4); a memory 3, 4 (p. 6 ll. 11-13); and a decoupling element 6 (p. 7 ll. 4-14; Figs. 2-3); wherein computations are passed to the second processor from the first processor through the decoupling element, such that the second processor executes computations passed from the first processor through the decoupling element (p. 18 ll. 15-24), and wherein the second processor receives data from and writes data to the memory, and wherein the coprocessor controller controls the activity of the second processor to ensure execution of the second processor is correctly ordered with respect to loads from memory, whereby the execution of computations by the second processor is decoupled from the operation of the first processor such that the second processor executes computations passed from the first processor through the decoupling element while the first processor is providing further instructions to the decoupling element (p. 19 ll. 8-26).

The invention claimed in claim 18 is directed to a method of operating a computer system comprising providing code for execution by a first processor 1 and a second processor 2 acting as coprocessor to the first processor (p. 6 l. 32 – p. 7 l. 7; Figs. 1-4); identifying a part of the code as providing a task to be carried out by the second processor (p. 23 ll. 8-15); passing information defining the task from the first processor to a decoupling element 6 (p. 23 l. 17 - p. 24 l. 2); and passing instructions derived from said information from the decoupling element to the second

processor and executing said instructions on the second processor, wherein the processing of said instructions by the second processor is decoupled from the operation of the first processor such that the second processor executes said instructions passed from the decoupling element while the first processor passes further information defining the task to the decoupling element (p. 19 ll. 8-26; Figs. 5-6).

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Issue 1: Whether claims 1-12, 14, 15, 17 and 18 are patentable under 35 U.S.C. 102(b) over U.S. Patent No. 5,708,830 to Stein.

ARGUMENT

Issue 1: Whether claims 1-12, 14, 15, 17 and 18 are patentable under 35 U.S.C. 102(b) over U.S. Patent No. 5,708,830 to Stein.

In section 2 of the final Office Action of February 6, the Examiner once again rejects claims 1-12, 14, 15, 17 and 18 under 35 U.S.C. 102(b) as being anticipated by U.S. Patent No. 5,708,830 to Stein. In their previous reply, Appellants had explained that Stein does not in fact disclose all of the limitations of the independent claims, noting that the claimed decoupling element is not in fact anticipated by Stein's "elements 6, 18 and 14" as this was an overly broad reading of the claim limitation in view of the plain language of the reference. In the present Action, the Examiner alleges to answer Appellants by now asserting that the claimed decoupling element is anticipated by Stein's "elements 6, 18, 14, 4, 10 and 12" – essentially taking a thin argument and stretching it even thinner. As previously, and thoroughly, explained, this is a completely erroneous reading of Stein.

Element 6 is "a writable control store 6 mapped into the memory area of the host" that the address bus communicates with (col. 2 ll. 54-56). "So far as internal operation of the coprocessor is concerned, the writable control store 6 operates only in read mode, addressable as 2K of 56 bit words, the seven blocks being enabled simultaneously for reading a word, identified by an eleven bit address from a sequencer 14, into a 56 bit wide pipeline latch 16. Sixteen bits of the output from the pipeline communicate with the bus DBUS, the remaining 40 bits of the output forming control signals for various components of the coprocessor as described further

below" (col. 3 ll. 3-11). Element 18 is a microprocessor that is also referred to as the intermediate processor, and which communicates with an intermediate bus DBUS that also connects with, among others, element 14. Element 14 is a sequencer which, in run mode, "issues addresses to the writable control store 6 under control of a clock signal, typically at 5 MHz, the address sequence being in accordance with a control signal received by the microprocessor. The 56 bit word selected from the control store 6 by an address from the microsequencer is latched by the pipeline latch 16 for one clock cycle. This word comprises control signals and instructions for the GAPP array 3 and its associated buffers and rotators and/or for the microprocessor 18 and/or for the microsequencer 14, as detailed above, which are acted on accordingly." (col. 5 ll. 48-58)

What is important to understand is that the system of Stein has two different modes of operation: a boot mode and a run mode. Instructions are loaded into the writable control store 6 only during the boot mode (col. 2 ll. 58-63), during which the sequencer is disabled (col. 5 ll. 31-32). During the run mode, when the host processor and the coprocessor run asynchronously, only data to be operated on is transferred to the coprocessor (col. 2 l. 63 – col. 3 l. 2), and this is the only mode during which the sequencer is enabled (col. 5 ll. 36-45).

Claim 1, on the other hand, recites that the second processor executes computations passed from the first processor through the decoupling element while the first processor is providing further instructions to the decoupling element. This is clearly not possible in the system of Stein because at least part of what the Examiner likens to the claimed decoupling element is disabled while instructions are being passed to the coprocessor (during the boot mode).

Applicants further note that the Examiner views the sequencer 14 not only as a subpart of the alleged decoupling element of Stein, but also assigns it double duty as the claimed coprocessor controller. This reading is clearly at odds with itself. The sequencer 14 is either a coprocessor controller that controls the activity of the second processor to ensure execution of the second processor is correctly ordered with respect to loads from memory, or it is a decoupling element through which computations are passed to the second processor from the first processor such that the second processor executes computations passed from the first

processor through the decoupling element and this execution of computations by the second processor is decoupled from the operation of the first processor - it cannot be both.

In the final Action, the Examiner attempts to address the above shortcomings of Stein by further invoking elements 4, 10, and 12. With regards to elements 10 and 12, "[d]ata transfer to and from the coprocessor during a run mode of the latter is handled through input and output first-in/first-out (FIFO) registers 10 and 12 connected between the bus IOBUS and an intermediate bus DBUS of the coprocessor, thus providing data buffers enabling the host processor and coprocessor to run asynchronously." [emphasis added] Element 4 is nothing but a buffer: "[t]he data lines of the memory blocks are connected to the bus IOBUS through a transceiver set 8, which in common with the buffer 4, is normally only enabled only when the coprocessor is in a "boot" mode (see below), for loading data and instructions to or downloading data and instructions from the writable control store." Thus, in accordance with the above discussion, the buffer 4 is operational only during the boot mode, whereas the registers 10 and 12 are operational only during the run mode. Thus, to reiterate, part of what the Examiner likens to the claimed decoupling element in the system of Stein is disabled while instructions are being passed to the coprocessor (during the boot mode), and thus the Examiner's perceived decoupling "elements 6, 18, 14, 4, 10 and 12" cannot possibly read on the claimed system wherein the second processor executes computations passed from the first processor through the decoupling element while the first processor is providing further instructions to the decoupling element.

The Examiner insistence to the contrary that "instructions are executed while instructions and data are provided to the FIFO's" is simply wrong. Even a cursory read of the very passage cited for support by the Examiner (at bottom of col. 2) makes this abundantly clear:

> The data lines of the memory blocks are connected to the bus IOBUS through a transceiver set 8, which in common with the buffer 4, is normally only enabled only when the coprocessor is in a "boot" mode (see below), for loading data and instructions to or downloading data and instructions from the writable control store. *Data transfer to and from the coprocessor during a run mode of the latter is handled through input and output first-in/first-out (FIFO) registers 10*

and 12 connected between the bus IOBUS and an intermediate bus DBUS of the coprocessor, thus providing data buffers enabling the host processor and coprocessor to run asynchronously.

This is further, and clearly, spelled out at col. 5 ll. 26 – 47:

The coprocessor has two alternative modes of operation. **In a boot mode ... the output address bus from the microsequencer 14 is disabled** and the buffer 4 and transceiver 8 are enabled so that the writable control store can be addressed by the host through bus PCADDR and the contents of the control store loaded or unloaded via the bus PCDATA. Additionally, all counters and **FIFOs are cleared**. To terminate the boot mode and enter a run mode... a decoder which resets the microsequencer 14 and re-enables its address output, **and disables the buffer 4** and register 8. Resetting the microsequencer clears its stack and sets its program counter to zero. Subsequent communication with the host is via the input and output FIFOs 10 and 12 **until the boot mode is again re-entered**.

Stein simply could not be clearer: it provides a system that switches between two different modes, boot and run, to either load instructions or to execute these instructions, respectively, but not both at the same time. The system of claim 1 clearly does not require such different modes of operation because of the presence of the decoupling element, which allows the first processor to pass computations to the second processor while the second processor is executing other computations. There is no such decoupled operation taking place in Stein.

Claims 2- 17 depend from claim 1. In view of the above discussion, it is submitted that claim 1 is allowable, and for this reason claims 2- 17 are also allowable.

Claim 18 is a method claim that corresponds to apparatus claim 1 and that includes the step of a second processor executing instructions passed from a decoupling element while a first processor passes further information defining the task to the decoupling element. As elaborated

upon above, this is a different method of operating than that disclosed by Stein, and Appellants respectfully submit that this claim is therefore also novel over Stein.

In view of all the above, Appellants respectfully submit that all pending claims are novel and nonobvious and request the Board to overturn the Examiner's rejection of the claims on appeal and pass the case to allowance.

Appellants further wish to challenge the finality of the Office Action of February 6, 2006, which the Examiner supports by alleging that "Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action." There are no new grounds of rejection presented in this Action, but rather slightly different portions of the same Stein reference being cited to by the Examiner in support of her rejections. The same reference is not a new grounds of rejection, and thus the Office Action of February 6, 2006 is not a proper final Action.

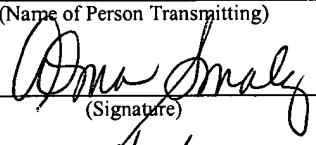
CONCLUSION

For the many reasons advanced above, Appellants respectfully contend that each claim is patentable and reversal of all rejections and allowance of the case is respectfully solicited.

I hereby certify that this correspondence is being deposited with the United States Post Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on

February 9, 2007
(Date of Transmission)

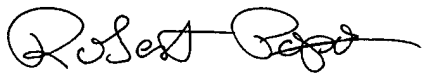
Alma Smalling
(Name of Person Transmitting)


(Signature)

2/9/07
(Date)

Attachments

Respectfully submitted,



Robert Popa
Attorney for Appellants
Reg. No. 43,010
LADAS & PARRY
5670 Wilshire Boulevard, Suite 2100
Los Angeles, California 90036
(323) 934-2300 voice
(323) 934-0202 facsimile
rpopa@ladasparry.com

Claims

1. A computer system, comprising:

a first processor;

a second processor for use as a coprocessor to the first processor;

a coprocessor controller;

a memory; and

a decoupling element;

wherein computations are passed to the second processor from the first processor through the decoupling element, such that the second processor executes computations passed from the first processor through the decoupling element, and wherein the second processor receives data from and writes data to the memory, and wherein the coprocessor controller controls the activity of the second processor to ensure execution of the second processor is correctly ordered with respect to loads from memory, whereby the execution of computations by the second processor is decoupled from the operation of the first processor such that the second processor executes computations passed from the first processor through the decoupling element while the first processor is providing further instructions to the decoupling element.

2. A computer system as claimed in claim 1, wherein the decoupling element is a coprocessor instruction queue, wherein computations are added to the coprocessor instruction queue by the first processor and consumed from the coprocessor instruction queue by the coprocessor.

3. A computer system as claimed in claim 1, wherein the decoupling element is a state machine, wherein information to provide computations to the second processor is provided to the state machine by the first processor, and computations are provided in an ordered sequence to the second processor by the state machine.

4. A computer system as claimed in claim 1, wherein the decoupling element is a third processor, wherein information to provide computations to the second processor is provided to the third processor by the first processor, and computations are provided in an ordered sequence to the second processor by the third processor.

5. A computer system as claimed in claim 1, wherein the second processor is configurable.

6. A computer system as claimed in claim 5, wherein the second processor is adapted to be configured in accordance with a configuration downloaded from the memory.

7. A computer system as claimed in claim 1, wherein the first processor is able to switch tasks during execution of computations by the second processor.

8. A computer system as claimed in claim 1, further comprising a buffer memory from which the second processor loads data and to which the second processor stores data, wherein the buffer memory is adapted to load data from the memory and store data to the memory.

9. A computer system as claimed in claim 8, wherein the memory is dynamic random access memory, and the buffer memory is adapted to load data from, or store data to, the buffer memory in bursts.

10. A computer system as claimed in claim 8, further comprising a second decoupling element, wherein memory instructions relating to movement of data between the buffer memory and the memory are passed to the buffer memory from the first processor through the second decoupling element, such that the buffer memory consumes instructions derived from the first processor through the second decoupling element, whereby the processing of memory instructions by the buffer memory is decoupled from the operation of the first processor.

11. A computer system as claimed in claim 10, wherein the second decoupling element is a buffer memory instruction queue, wherein memory instructions are added to the buffer memory instruction queue by the first processor and consumed from the buffer memory instruction queue by the buffer memory.

12. A computer system as claimed in claim 11, wherein the second decoupling element is a state machine, wherein information to provide memory instructions to the buffer memory is provided to the state machine by the first processor, and memory instructions are provided in an ordered sequence to the buffer memory by the state machine.

13. A computer system as claimed in claim 10, wherein the second decoupling element is a fourth processor, wherein

information to provide memory instructions to the buffer memory is provided to the fourth processor by the first processor, and memory instructions are provided in an ordered sequence to the buffer memory by the fourth processor.

14. A computer system as claimed in claim 8, further comprising a synchronisation mechanism to synchronise transfer of data between the buffer memory and the memory with execution of computations by the second processor.

15. A computer system as claimed in claim 14, wherein the synchronisation mechanism is adapted to block execution of computations by the second processor on data which has not yet been loaded to the buffer memory from the memory, and is adapted to block execution of memory instructions for storage of data from the buffer memory to the memory where relevant computations have not yet been executed by the second processor.

16. A computer system as claimed in claim 15, adapted such that when execution of computations or memory instructions is blocked by the synchronisation mechanism, other computations or memory instructions which are not blocked by the synchronisation mechanism may be executed.

17. A computer system as claimed in claim 1, wherein the first processor is the central processing unit of a computer device.

18. A method of operating a computer system, comprising:

providing code for execution by a first processor and a second processor acting as coprocessor to the first processor;

identifying a part of the code as providing a task to be

carried out by the second processor;

passing information defining the task from the first processor to a decoupling element; and

passing instructions derived from said information from the decoupling element to the second processor and executing said instructions on the second processor, wherein the processing of said instructions by the second processor is decoupled from the operation of the first processor such that the second processor executes said instructions passed from the decoupling element while the first processor passes further information defining the task to the decoupling element.

There is no evidence submitted with the present Brief on Appeal.

There are no other appeals or interferences related to the present application.